Method of Modifying a Volume Mesh Using Sheet Insertion

INVENTOR(S):

Michael J. Borden
6200 Eubank Blvd. NE
Apt. 1423
Albuquerque, New Mexico 87111

Jason F. Shepherd
9428 De Vargas Loop NE
Albuquerque, New Mexico 87109

METHOD OF MODIFYING A VOLUME MESH
USING SHEET INSERTION

Cross-Reference

[01] This application claims the benefit of U.S. provisional application, Ser. No. 60/390,957, filed June 24, 2002.

[02] The U.S. Government has rights in this invention pursuant to Department of Energy Contract DE-AC94AL-85000 with Sandia Corporation.

Technical Field

[03] The present invention relates generally to computer modeling of physical systems. It particularly relates to computer modeling allowing modification of a finite element volume mesh using dual generation and sheet insertion.

Background of the Invention

[04] With the advancement of computer technologies and understanding of basic physical phenomena or systems (e.g., engine operation, fluid flow, heat transfer, structural stress and strain analysis, etc.), three-dimensional (3D) computer simulation has become more of an important feature in physical system development, analysis, and evaluation. The computer simulation (modeling) often involves the building of a finite element mesh (collection of discrete set of points defined as nodes) to model the physical system. The accuracy of finite element mesh generation is related to the geometric complexity (including representing the physical system by a set of mathematical equations) of the physical system including the number of finite elements in the mesh, the order of those elements, and the quality of those finite elements.

[05] A number of mesh-generating algorithms (e.g., parametric mapping, Dicer algorithm, Paving algorithm, Whisker-Weaving algorithm, sweeping algorithm, etc.) have been developed to

attempt to generate high-quality meshes (including volume meshes) with greater accuracy and reduced user interaction for generating the mesh. However, each algorithm has its own set of strengths and weaknesses, and therefore may only be suitable for a particular geometry while being ineffective for another. Therefore, there is still a need to generate high-quality meshes for all types of geometry, including hexahedral volume meshes, that are robust, accurate, and reduce user interaction time. Additionally, modification of a volume (3D) mesh is an important feature to improving mesh quality by allowing insertion of additional elements (e.g., introduce new elements to form a more complex geometry) into the mesh to generate a more detailed volume mesh and more accurate and faster analysis results.

[06] As described herein, the generation of a dual (for a volume mesh) within a dual space may be an effective tool for producing a high-quality volume mesh for three-dimensional elements (objects) by providing an alternative geometric representation of the volume mesh and more clearly defining global connectivity constraints for the mesh. Advantageously, the dual of a mesh may be generated, edited, and then converted back to a volume mesh to improve analysis results. It is noted that terms used within the specification, in accordance with embodiments of the present invention, will be defined within the specification and further definition may be found within the Glossary of Terms in Appendix A. FIGs. 1A, 1B illustrate the process for generating a dual of a 3D element as found in the prior art. FIG. 1A shows a stack (column) 100 of 3D elements (mesh) in primal space (e.g., hexahedral elements). Each hexahedral element of the stack 100 includes six quadrilateral faces 108 and eight nodes 110 formed from three edges 112. It is noted that stack 100 may form the complete volume mesh. A dual 115 of the volume elements (mesh) 100 may be generated by connecting opposing faces of a hexahedral element using a (volume) chord 102 (see Appendix A for glossary of terms) as shown in FIG. 1B. As shown in FIG. 1B, chord 102 (a dual volume chord) connects the opposite edges for a stack of hexahedral elements 114, 116, 118, 120. In the dual space generated, chord 102 is equivalent (the dual) to the row of hexahedral elements 114, 116, 118, 120 in the primal space.

[07] The generation of the dual may continue as shown in FIG. 1B as more opposite faces of the hexahedral elements 114, 116, 118, 120 are connected using further chords (e.g., 101, 103, 105, 107, 109). The chords are generated with adherence to the following rules: 1) a chord

that begins on a boundary must terminate on the boundary, or 2) a chord may form an internal closed loop.

[08]   To help complete the dual 115, a twist plane 202 may be generated as shown in FIG. 2A (from the prior art) that carries a chord 102 along an intersecting edge. The twist plane 202 may be a continuous, three-dimensional surface which adheres to the following rule: twist planes may be nowhere tangent or coplanar. FIG. 2A found in the prior art shows three intersecting twist planes 202, 204, 206 that define a 3D cell region (hexahedral element) 208. Three-dimensional (3D) cell region 208 may be defined as an n-sided polyhedron with the faces formed by individual twist planes 202, 204, 206 that carry (formed from) chords 102, 101, 109, respectively (see glossary in Appendix A). As shown in FIG. 2B from the prior art, a centroid 216 may be formed from the three intersecting chords 101, 102, 109 generated from the intersecting twist planes 202, 204, 206 where the intersecting chords include one 3D cell region (hexahedral element) 208. FIG. 3 found in the prior art shows a twist plane 302 in a hexahedral mesh 300 that may be used to generate a sheet of hexahedral mesh elements for extraction to modify the mesh 300. As shown in FIG. 2B, every 3D cell region 208 includes a single node (e.g., node 210) from the original stack 100. Cell region 208 is equivalent (the dual) to node 210 within the dual space generated. Also, centroid 216 is equivalent (the dual) to hexahedral element 114 within the dual space generated. Also, Table 1 in Appendix B shows the relationship between the original surface elements and dual entities in three dimensions.

[09]   Therefore, due to the disadvantages of current volume meshing algorithms, there is a need to provide a computer modeling technique that uses duals to modify hexahedral volume meshes while maintaining accuracy, reduced user interaction time, and high quality of the resulting meshes to generate a more detailed hexahedral volume mesh.

<div align="center">Summary of the Invention</div>

[10]   The method and machine-readable medium of the present invention overcome the previously mentioned problems by providing a technique to modify a hexahedral finite element volume mesh using dual generation and sheet insertion. After generating a mesh of the volume, a predetermined algorithm may be followed to modify (refine) the volume mesh of hexahedral elements. The predetermined algorithm may include the steps of locating a sheet of hexahedral mesh elements, determining a plurality of hexahedral elements within

the sheet to refine, shrinking the plurality of elements, and inserting a new sheet of hexahedral elements adjacently to modify the volume mesh. Additional features of the invention include using a mesh cutting technique to modify a volume mesh.

Brief Description of the Drawings

[11] Fig. 1A is a block diagram of exemplary volume elements found in the prior art.

[12] Fig. 1B is a block diagram of exemplary dual generation found in the prior art.

[13] Fig. 2A is a block diagram of an exemplary dual of a volume element generated using twist planes found in the prior art.

[14] Fig. 2B is a block diagram of exemplary generated dual elements found in the prior art.

[15] Fig. 3 is a block diagram showing a twist plane found in the prior art.

[16] Fig. 4 is a block diagram showing exemplary elements employed in a sheet insertion algorithm in accordance with embodiments of the present invention.

[17] Fig. 5A is a block diagram of an exemplary generated dual showing a sheet of polyhedral elements in accordance with embodiments of the present invention.

[18] Fig. 5B is a block diagram of an exemplary generated dual showing a group of polyhedral elements to be refined in accordance with embodiments of the present invention.

[19] Fig. 5C is a block diagram of an exemplary modified volume mesh from a generated dual after sheet insertion in accordance with embodiments of the present invention

[20] Fig. 6A is a block diagram of an exemplary existing volume mesh in accordance with embodiments of the present invention.

[21] Fig. 6B is a block diagram of an exemplary generated sheet of polyhedral elements to be inserted into a volume mesh in accordance with embodiments of the present invention.

[22] Fig. 6C is a block diagram of an exemplary modified volume mesh after sheet insertion in accordance with embodiments of the present invention.

[23] Fig. 7 is a flow process diagram of a sheet identification algorithm to identify a sheet of polyhedral elements within a volume mesh in accordance with embodiments of the present invention.

[24] Fig. 8 is a flow process diagram of a sheet insertion algorithm to modify a volume mesh in accordance with embodiments of the present invention.

[25] Fig. 9 is a flow process diagram of a mesh cutting algorithm to modify a volume mesh in accordance with embodiments of the present invention.

[26] Fig. 10A is a block diagram of an exemplary existing volume mesh in accordance with embodiments of the present invention.

[27] Fig. 10B is a block diagram of an exemplary volume to be inserted into an existing volume mesh in accordance with embodiments of the present invention.

[28] Fig. 10C is a block diagram of an exemplary volume inserted into an existing volume mesh with nodes moved to the surface of the intersection in accordance with embodiments of the present invention.

[29] Fig. 10D is a block diagram of a modified volume mesh with an exemplary volume removed after insertion in accordance with embodiments of the present invention.

[30] Fig. 10E is a block diagram of an exemplary modified volume mesh after mush cutting in accordance with embodiments of the present invention.

[31] Fig. 11 is a flow process diagram of a transition path insertion algorithm to modify a volume mesh in accordance with embodiments of the present invention.

[32] Figs. 12A-12C are block diagrams showing exemplary performance of transition path refinement in accordance with embodiments of the present invention.

[33] Fig. 13 is a block diagram of an exemplary system performing volume mesh modification in accordance with embodiments of the present invention.

[34] Figs. 14A-14B are block diagrams showing exemplary transition paths in accordance with embodiments of the present invention.

[35] Fig. 15 is a block diagram showing an exemplary transition path of a volume mesh in accordance with embodiments of the present invention.

Detailed Description

[36] In accordance with embodiments of the present invention, a sheet insertion algorithm may be followed (as executed by a machine-readable medium) to modify a finite element volume

mesh using a dual of a volume mesh (e.g., a three-dimensional brick structure). Fig. 4 shows a block diagram including exemplary elements employed in a sheet insertion algorithm in accordance with embodiments of the present invention. A volume mesh 400 may include a stack of hexahedral elements (hexahedrons) 402, 403 where each hexahedral element includes six quadrilateral faces 404 and eight predetermined nodes 406 with each node formed at an intersection of three edges 408. As shown in FIG. 4, a dual 409 of the stack of hexahedral elements 402 may be generated (following the steps of FIG. 1B) using volume chords 416, 417 where an intersection of multiple chords forms volume centroids 418, 419 in the middle of hexahedral elements 402, 403. Advantageously, a twist plane 412, defined to start from mesh edge 408 including chord 420 intersecting surface centroids 422, 424, may be used to represent a sheet of hexahedral elements from dual 409. Surface centroids (the intersection of two or more surface chords) 422, 424 may represent the end points of volume chords 416, 417. Volume chord 416 may be selected as the chord lying within (along an intersecting edge with other twist planes as shown in FIG. 2A) twist plane 412 to define the sheet of hexahedral mesh elements starting with hexahedral element 402.

[37]   As shown in FIG. 5A, one or more sheets 502 of hexahedral elements may be generated from volume mesh 500, starting with initial hexahedral element 504, using the flow process of FIG. 7. The flow process uses the condition of mesh configuration that neighboring hexahedral elements may share one face and lining up hexahedral elements up so that each element has two neighboring elements that are attached to opposing faces will generate columns of hexahedral elements. Following the flow process, at step 702 the initial hexahedral element 504 in the sheet 502 is identified along with the faces of element 504. At step 704, neighboring element 506 may be identified using the shared face 508 between initial element 504 and neighboring element 506. At step 706, the step of 704 is continued until all neighboring elements (508, 510, 512, 514, 516, 518, 520, 522, 524, 526, 528) in a column are identified to form one or more sheets 502 of hexahedral elements. Advantageously, one or more sheets 502 may be represented one or more twist planes, each plane located along a volume chord lying within the plane.

[38]   In accordance with embodiments of the present invention, modification of a volume mesh 500 by sheet insertion (auto refinement) may be performed using the flow process of FIG. 8

as shown in FIGs. 5A-5C. At step 802, one or more sheets 502 of hexahedral elements may be generated (using the flow process of FIG. 7) as shown in FIG. 5A. At step 804, a group of hexahedral elements within one or more sheets 502 to be refined may be determined (defined) as shown by exemplary group 530 (including elements 504, 506, 508, etc.) in FIG. 5B. A predetermined algorithm may be followed to determine the particular group 530 of hexahedral elements to refine within one or more sheets 502. Advantageously, each (hexahedral) element in one or more sheets 502 may be examined during this process. Each element (e.g., 504) of the one or more sheets 502 includes a set of opposing faces (top and bottom faces) that are not shared by any other element in the one or more sheets 502. The distance (d) between these two faces may be determined and recorded for each element in the one or more sheets 502 including recording of the element with the shortest distance between opposing, non-sharing faces. The distance measurement for every other element in the one or more sheets 502 may be compared with the shortest distance to generate a ratio for every other element in the one or more sheets 502. Thereafter, the generated ratio may be compared with a predetermined ratio threshold (e.g., user specified) and all elements with a ratio satisfying this threshold (e.g., equal to or greater) may be placed in the group 530 of hexahedral elements to be refined.

[39] At step 806, after group 530 has been determined, a new sheet 532 of hexahedral elements may be inserted into mesh 500 to produce modified volume mesh 538 as shown in FIG. 5C. Advantageously, new sheet 532 may be inserted using the process of pillowing. The identified group 530 of hexahedral elements to be refined are shrunk (shrink region) wherein the exterior nodes of the shrink region (set) 530 are moved (outwards) while retaining a copy of each in the original position 542. Thereafter, the elements (refinement group) of the shrink region 530 are completely separated from the surrounding mesh 500 by replacing the nodes of the surrounding hexahedral elements that are on the boundary of the shrink set 530 with the corresponding copied nodes at the original position 542. This step forms a void 534 between the replacement, copied nodes in the original position 542 and the actual shrink set 530. Thereafter, new sheet 532 may be inserted to fill void 534 as shown in FIG. 5C.

[40] Advantageously, volume mesh 500 may be initially generated using the sheet generation algorithm of FIG. 7 and step 802 of FIG. 8 to generate a plurality of sheets of hexahedral

elements along volume chords and associated twist planes (as shown in FIGs. 4-5). Thereafter, the generated mesh may be modified using further steps of the flow process of FIG. 8 (steps 804, 806) to determine (identify) a group of elements to refine, and insert a new sheet using the pillowing process to generate the modified volume mesh 538 as shown in FIG. 5C.

[41]   Additionally, FIGs. 6A-6C show an exemplary modification of a volume mesh 600 using the flow process of FIG. 8 in accordance with embodiments of the present invention. Mesh 600, prior to sheet insertion, is shown in FIG. 6A and a shrink region (group of elements to be refined) 606 including multiple columns of (hexahedral) elements is identified (determined). In FIG. 6B, a new sheet 602 to be inserted is shown, and then in FIG. 6C the modified volume mesh 604 is shown wherein sheet 602 has been inserted to fill a void surrounding the shrink region 606 to produce the modified mesh 604. Sheet 602 maintains the all-hexahedral connectivity of mesh 604 and may be inserted to improve the quality of the mesh by producing a more uniform (geometric complexities resolved) mesh. Advantageously, the process of FIG. 8 may be used for (local) feature refinement of a mesh using a plurality of different elements to identify the shrink region (group of elements to refine) including a surface, line, or point within the mesh.

[42]   In accordance with embodiments of the present invention, a variation of sheet insertion (mesh cutting) may be performed to modify a volume mesh using the flow process of FIG. 9 as shown in FIGs. 10A-10E. FIG. 10A shows an existing volume mesh 1000 prior to mesh cutting with nodes 1006. At step 902, a (new) volume (e.g., cylinder) 1002 may be inserted into the original mesh 1000 (as shown in FIG. 10B) forming an intersection 1004 at the surface 1005 of the volume 1002 between the surrounding mesh 1000 and the volume 1002.

[43]   At step 904, the nodes 1006 of the elements 1008 at the intersection 1004 may be moved from their original position (as shown in FIG. 10A) to the surface 1005 of the volume 1002 (as shown in FIG. 10C). Additionally, new layers of meshed elements 1008 may be added (re-meshing) at the surface 1005 as shown in FIG. 10C. FIG. 10D shows the mesh 1000 with the inserted volume 1002 removed while the elements 1012 from inside volume 1002 remain inserted. Thereafter, at step 906, the inserted elements 1012 (coming from inside

volume 1002) are removed to produce modified volume mesh 1014 with cut-out region 1016 as shown in FIG. 10E.

[44]   In accordance with embodiments of the present invention, the volume mesh modification algorithm described herein may include a transition path insertion algorithm to insert a new sheet of elements along a transition path (path between linking surfaces of the mesh which may lessen quality of the mesh) of the mesh. Firstly, a transition path may be defined in the mesh by using a shortest weighted path algorithm. The path may be defined by a set of linked nodes that form a line that passes through the volume mesh. The terminating ends of the path are located on the linking surfaces of the volume, and the path may be found using a shortest weighted path algorithm based on a predetermined algorithm (e.g., Dijkstra's algorithm). The distance of the path may be weighted to ensure the following: 1) minimize the number of nodes in the path, 2) keep the path as straight as possible, and 3) keep the path as far from non-terminating linking surfaces as possible. Advantageously, these objects may improve the quality of the resulting mesh by keeping the number of hexahedral elements in the transition to a minimum and allows as much room as possible for the elements in the transition to be smoothed.

[45]   In accordance with the weighted path algorithm, the weighted distance of a node may be defined as $dist = (p + 1) + t + (w_{max} - w_{node})$, where $dist$ = weighted distance of the node; p = weighted distance of previous node in the path; t = 0 if the node path does not turn, or 1 otherwise; w_{node} = weight of node; and w_{max} = maximum weight of all nodes

[46]   Advantageously, the distance a node is from the linking surfaces determines its weight value. A node on a linking surface may be weighted zero and the node furthest from any linking surface may be weighted w_{max}. Therefore, the value $(w_{max} - w_{node})$ may be added to the distance to meet the objective of moving the path away from the surfaces. The value t may be determined by the "straightness" of the path. In accordance with embodiments of the present invention, FIG. 14A shows a path (shown by arrows 1401) with consecutive nodes 1402 that do not turn along arrows 1405 and alternatively, FIG. 14B shows a path (shown by arrows 1404) with consecutive nodes 1406 that do turn.

[47]   As shown in Appendix C, the shortest weighted path algorithm may include a breadth first search that proceeds through steps 1-9. At step 7, if the search group is empty before the

end node is reached then the search group may be disjoint and a path between the start and end nodes cannot be found so the algorithm may return a failure.

[48]   As the shortest weighted path algorithm in Appendix C is executed, each node that has been visited may hold a pointer to the node immediately before it in the path. Therefore, once the end node may be reached, the path may be found by starting with the end node and following the pointers back to the start node. FIG. 15 shows a transition path 1502 that may found through an existing volume mesh 1500 using the shortest weighted path algorithm of Appendix C in accordance with embodiments of the present invention.

[49]   Once the transition path is found (determined), the path may be projected through the volume mesh until a target surface is reached. Figs. 12A-12C are block diagrams showing exemplary performance of transition path extraction and FIG. 11 shows a flow process diagram of transition path insertion in accordance with embodiments of the present invention. At step 1102, the transition path may be found (determined) using the shortest weighted path algorithm of Appendix D. As shown in FIG. 12A, a transition path may be found and then projected through an exemplary volume mesh 1200 to form a sheet 1202 of nodes on a target surface 1204 that may be used to define a shrink region.

[50]   At step 1104, the shrink region of hexahedral elements may be determined as the elements 1206 having a face lying on the sheet as shown in FIG. 12B. After the shrink region 1206 of elements are determined from sheet 1202, a void 1208 may be formed using the pillowing process described herein. FIG. 12B shows the determined shrink region 1206 of elements and void 1208 formed within volume mesh 1200. Thereafter, at step node 1106, a (new) sheet 1210 of hexahedral elements may be inserted to fill void 1208 and produce modified volume mesh 1212. Sheet (transition elements inserted) 1210 may include multiple valent nodes (nodes leading to lower quality of mesh) whose presence is minimized by the objectives of the shortest weighted path algorithm followed in Appendix C.

[51]   As described herein, the volume mesh modification algorithms (including sheet and transition path extraction) described herein may be performed by a computer system using a machine-readable medium. Fig. 13 is a block diagram of an exemplary system performing dual generation in accordance with embodiments of the present invention. System 1300 includes an input device 1302, processing device 1304, display 1306, and storage media 1308. Advantageously, processing device 1304 may automatically execute the volume

modification algorithm (including sheet insertion, mesh cutting, and/or transition path refinement as shown in FIGs. 8-9, 11) by retrieving a volume mesh from storage media 1308, and display the resulting mesh on display 1306. Alternatively, one or more of the individual steps of the volume mesh modification algorithm may be performed in response to commands received via input device 1302.

[52] A plurality of advantages may be provided in accordance with embodiments of the present invention including a volume mesh modification method (including sheet insertion, mesh cutting, and transition path insertion) that allows mesh elements (e.g., hexahedrons) to be modified without regard to neighboring elements enabling independent editing of mesh elements. Additionally the volume mesh modification algorithm enables generation of a high-quality resulting mesh by recognizing global connectivity information (e.g., local self-intersections and self-tangencies of twist planes - volume chords) regarding the mesh.

[53] Although the invention is primarily described herein using particular embodiments, it will be appreciated by those skilled in the art that modifications and changes may be made without departing from the spirit and scope of the present invention. As such, the method disclosed herein is not limited to what has been particularly shown and described herein, but rather the scope of the present invention is defined only by the appended claims.

Appendix A

[54] Glossary of Terms:

[55] Dual: Geometric copy of object (mesh) created in a dual space using volume chords and twist planes

[56] Chord: A (volume) chord is a collection of one or more segments. It is the dual of a row (column) of hexahedral elements in a volume (mesh). A chord can start at a face (surface centroid) and terminate at another face (i.e. a chord with two end points), or it can loop back and close on itself (an internal closed loop). A chord must contain at least one segment. A chord may only be defined on a twist plane at the intersection of two twist planes.

[57] 3-Cells: An intersection of twist planes, each intersection defining a (volume) chord, that define an n-sided polyhedron in dual space. The 3-cell is dual to a node, meaning one three cell exists for every node in the volume mesh.

[58] Twist Plane: A continuous, three-dimensional surface used to represent a sheet of hexahedral mesh elements. The twist plane may be defined by the end points of two volume chords on a mesh boundary.

[59] Centroid: a centroid is the intersection of two or more chords. For every intersection of chords in the dual, there exists one hexahedral element in the volume (mesh). A centroid is the dual of a hexahedral element.

[54]   Appendix B

Table 1

Mesh Entity	Dimension	Dual Entity	Dimension
Hex Element	3	Centroid	0
Face	2	Chord	1
Edge	1	2-cell	2
Node	0	3-cell	3

Appendix C - Shortest Weighted Path Algorithm

Step 1: Define the search group

Step 2: Define the start and end node of the path;

Step 3: Weight all nodes in the search group;

Step 4: Set distance of all nodes to maximum integer value;

Step 5: Set (start node).dist = 0;

Step 6: Set (start node).path = NULL;

Step 7: Search for path

        While end node not found

           If search group is empty

           Return failure;

           N = smallest unknown distance node;

           if N == end node

           goto Step 8;

           else

           Remove N from search group;

           Get nodes, M, adjacent to N;

           For each node M

           if {N.dist + 1 + M.t + (wmax – M.wnode)} < M.dist

               M.dist = N.dist + 1 + M.t + (wmax – M.wnode);

               M.path = N;

           end For;

        end While;

Step 8: Store path in list, L

        N = end node;

      While N.path != NULL

        L.append(N);

        N = N.path;

      end While

Step 9: return L;